Edge of Chaos Mapped by Euler Identity: Closed-Form Lyapunov Rim, Coupled Map Lattices, Dissipative Structures, and a Noether Sheaf-Theoretic Topos for Complexity

Douglas C. Youvan

doug@youvan.com

www.youvan.ai

September 29, 2025

Why do very simple update rules generate rich structure? We answer with a single, closed-form scaffold we call the Euler edge of order. Start with the logistic map and its r=4 conjugacy to angle doubling on the unit circle. Using Euler's identity we write the circle variable as z = e^(i*2*pi*theta) so the dynamics are z -> z^2. For a nearest-neighbor diffusive lattice built from this generator, the transverse growth rate of a Fourier mode k is lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|). At r = 4, the circle-doubling gives lambda_map(4) = ln(2). The edge of order is the zero set lambda_perp = 0, which yields a closed-form ridge in (k, eps) where ordered and chaotic tendencies balance. A simple, interpretable complexity score that concentrates on this ridge is C(r, eps) = h_top(r) * sech((lambda_map(r) + ln(|1 - 2*eps|))/lambda0). We treat the polar parameter annulus (angle k, radius eps) as a small topos: objects are patches, morphisms are edge-preserving maps, and sheaves carry fields such as complexity, Lyapunov sign, symmetry, and "currents" derived from gradients of C. Five figures tell the whole story: the base complexity universe with its luminous rim, the rim as a manifold with local sharpness, the Euler skeleton z -> z^2, the ordered vs chaotic partition, and a flow field showing how complexity accumulates near the edge.

Keywords: logistic map, angle doubling, Euler identity, Lyapunov exponent, edge of chaos, dissipative structures, coupled map lattice, complexity functional, topos, sheaves, symmetry, Noether balance, univalence, Mandelbrot analogy, cellular automata analogy.

A collaboration with GPT-5. CC4.0. 44 pages.

Outline

1. **Abstract**
   One paragraph: problem, method, main equations, visual result, significance.

2. **Background and Motivation**
   2.1 Simplicity -> complexity (Wolfram, Mandelbrot, Prigogine).
   2.2 Euler framing of r=4: z = e^(i*2*pi*theta), z -> z^2.
   2.3 Why a topos: gluing local truths along an edge.

3. **Model and Closed-Form Diagnostics**
   3.1 Generator and coupling: x_{i}^{n+1} = (1 - eps)*f_r(x_i^n) + (eps/2)*(f_r(x_{i-1}^n) + f_r(x_{i+1}^n)), f_r(x)=r*x*(1-x).
   3.2 Transverse spectrum: lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|).
   3.3 Edge ridge and complexity: eps*(r) = (1 - exp(-lambda_map(r)))/2;
   C(r, eps) = h_top(r) * sech((lambda_map(r) + ln(|1 - 2*eps|))/lambda0).

4. **The Eclipse Topos: Objects, Morphisms, Sites**
   4.1 Objects: patches U in the polar annulus; rim E = {lambda_perp = 0}.
   4.2 Morphisms: phase shifts (S^1), radial moves, edge-preserving maps.
   4.3 Covers: interior, exterior, and edge covers; gluing conditions.

5. **Sheaves and Invariants**
   5.1 Sheaf C: local stats of the complexity field.
   5.2 Sheaf L: Lyapunov sign classifier (ordered/edge/chaotic).
   5.3 Sheaf S: local symmetry stabilizers; circle doubling action.
   5.4 Currents J from grad(C); balance identities on vs off the rim.

6. **Results (Visual Narrative)**
   Fig. 1 Base complexity universe (polar heatmap of C with rim E).
   Fig. 2 Rim as manifold (marker size ~ |d lambda_perp / d eps| on E).
   Fig. 3 Euler skeleton on the unit circle with arrows z -> z^2.
   Fig. 4 Ordered (lambda_perp < 0) vs chaotic (lambda_perp > 0) regions; rim is the exact separator.
   Fig. 5 Complexity flow field: quiver of grad(C) mapped onto the disk.

7. **Symmetry and Noether-with-Ports (Open Systems)**
   7.1 From symmetry sections to currents J_sigma; div(J_sigma) = source - sink.
   7.2 Conservation on the rim (closed boundary) vs balance away from the rim (open).

8. **Toward Resource-Aware Univalence**
   8.1 Equivalence that preserves (lambda_perp, C) within graded tolerances.
   8.2 Paths allowed only along resource-respecting morphisms.

9. **Discussion**
   9.1 Relation to CA phase diagrams and edge-of-chaos work.
   9.2 Quadratic-family analogy: iteration geometry meets stability.
   9.3 Generalizations: 2D lattices, heterogeneous couplings, colored noise.

10. **Methods**
    Parameter grids, numerical checks for lambda_map(r), rendering notes for Figs. 1–5.

11. **Limitations and Future Work**
    Non-diffusive couplings, stacks over the rim, categorical Noether theorems for open systems.

12. **Conclusion**
    The eclipse rim as a minimal, closed-form scaffold where simplicity breeds structure; a topos that glues local truths along that rim.

References

Source Code

1. Abstract

Why do simple local rules yield rich global structure? We answer with a closed-form scaffold that makes the edge of order both solvable and visible. Starting from the logistic generator and its r=4 conjugacy to angle doubling on the unit circle (z = e^(i*2*pi*theta), dynamics z -> z^2), we build a nearest-neighbor diffusive lattice and analyze transverse stability in Fourier mode k. The key equation is the transverse exponent lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|), which at r = 4 uses the exact lambda_map(4) = ln(2). **The edge of order** is the zero set lambda_perp = 0, yielding the analytic ridge eps*(r) = (1 - exp(-lambda_map(r)))/2 (so eps*(4) = 1/4 at k = pi). To quantify structured richness we introduce a one-line score C(r, eps) = h_top(r) * sech((lambda_map(r) + ln(|1 - 2*eps|))/lambda0), which concentrates on the edge without ad hoc tuning. Our visual result is a five-figure "Eclipse Topos": (1) a polar heatmap of C with a luminous rim, (2) the rim as a manifold with local sharpness, (3) the Euler skeleton z -> z^2, (4) a sign partition of ordered vs chaotic regions, and (5) a complexity flow field. Significance: this minimal apparatus unifies Wolfram-style simplicity, Lyapunov diagnostics, Prigogine's open-system intuition, and the quadratic (Mandelbrot/Julia) skeleton, while providing a resource-aware categorical setting to glue local truths along the edge.

2. Background and Motivation

2.1 Simplicity -> complexity (Wolfram, Mandelbrot, Prigogine)

The thesis is that minimal generators can produce maximal variety. In Wolfram's 1D cellular automata, a 1-bit state, radius-1 neighborhood, and a tiny lookup table already yield long-lived motifs, gliders, and computation. The logistic family is similarly spare: x_{n+1} = r*x_n*(1 - x_n) uses one parameter and a quadratic nonlinearity, yet spans fixed points, period-doubling cascades, windows, and chaos. Mandelbrot's quadratic map z -> z^2 + c shows the same lesson in the complex plane: iterate a simple polynomial, and a frontier of unending detail appears. Prigogine adds the open-systems lens: drive plus dissipation can stabilize **dissipative structures** at far-from-equilibrium conditions, so pattern is not an accident of initial conditions but a maintained balance between injection and loss.

Our program keeps the generator simple and makes openness explicit. We couple nearest neighbors (diffusion = exchange with surroundings) and assess stability and richness with closed-form diagnostics. The edge where order and chaos trade off is not an article of faith; it is the zero set of a computable exponent. The figures are not illustrations of a story; they *are* the story the simple rule tells when it is gently opened to its environment.

2.2 Euler framing of r=4: z = e^(i*2*pi*theta), z -> z^2

At r = 4, the logistic map is conjugate to angle doubling. Let x = sin^2(pi*theta) with theta in [0,1). Then
x_{n+1} = 4*x_n*(1 - x_n) corresponds to theta_{n+1} = 2*theta_n (mod 1). Writing z = e^(i*2*pi*theta) puts the dynamics on the unit circle as z -> z^2. Two consequences matter:

1. The invariant density at r = 4 is explicit, and the (map) Lyapunov exponent is exact: lambda_map(4) = ln(2).
   Intuitively, each iterate doubles the angular phase, so average stretching is a factor of 2 per step.

2. Fourier modes are natural coordinates. When we couple sites diffusively, perturbations decompose as exp(i*k*i). The coupling contributes a factor (1 - eps + eps*cos(k)) to each mode, so the *transverse* growth rate separates cleanly:
   lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|).
   At r = 4, this becomes ln(2) + ln(|1 - eps + eps*cos(k)|), turning the edge condition lambda_perp = 0 into a simple analytic curve in (k, eps).

2.3 Why a topos: gluing local truths along an edge

The eclipse-like rim where lambda_perp = 0 is a boundary between two kinds of truth: inside, transverse perturbations decay (ordered/coherent); outside, they grow (chaotic/disordered). Each neighborhood of the parameter annulus carries valid but *local* statements: “here, this mode is stable,” “here, complexity peaks,” “here, symmetry breaks.” A topos viewpoint formalizes how such local propositions are organized, compared, and glued:

- **Objects:** open patches U of the polar parameter space (k, eps).
- **Sheaves:** assignments U -> data(U) that are locally defined and agree on overlaps: complexity statistics C, Lyapunov sign partitions, symmetry stabilizers, and derived “currents” from grad(C).
- **Covers and gluing:** interior and exterior covers meet on the rim E = {lambda_perp = 0}; sections that match on overlaps glue to global statements about the edge.

This logic is *resource-aware*: morphisms are required to respect the edge and the diagnostics that define it (e.g., preserve lambda_perp sign and the profile of C within tolerances). In practice, that means equivalence is not merely topological sameness but sameness that keeps the thermodynamic scaffolding intact. The benefit is conceptual and practical: proofs transport along admissible maps, visual evidence (the five figures) becomes structured data in sheaves,

and the “luminous rim” functions as the site where local truths consistently meet to make a global one.

3. Model and Closed-Form Diagnostics

3.1 Generator and coupling

We use the quadratic logistic generator f_r(x) = r*x*(1 - x) and couple identical sites on a 1D ring by nearest-neighbor diffusion **after** applying f_r (standard coupled-map lattice form):

x_i^{n+1} = (1 - eps)*f_r(x_i^n) + (eps/2)*(f_r(x_{i-1}^n) + f_r(x_{i+1}^n)).

Here i indexes sites, n indexes time, r controls local nonlinearity, and eps in [0,1] controls exchange with neighbors. This choice encodes “openness” (Prigogine flavor) without external reservoirs: each site exports/receives updated activity to/from its neighbors. Two modeling decisions make the later algebra clean:

1. **Post-map coupling:** diffusively mix f_r(x) (not raw x). This diagonalizes neatly in Fourier space.
2. **Homogeneous synchronized orbit:** linearize about x_i^n = x^n for all i. This background shares the single-site statistics (invariant density, Lyapunov) of the uncoupled map.

3.2 Transverse spectrum

Let delta_i^n be a small perturbation around the synchronized orbit. Linearizing and expanding in Fourier modes delta_i^n ~ exp(i*k*i) gives the multiplicative factor per step:

M(k; r, eps, x^n) = f_r'(x^n) * (1 - eps + eps*cos(k)).

Taking a long-time average of ln|M| yields the transverse Lyapunov exponent:

lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|).

The term lambda_map(r) = lim_{N->inf} (1/N) sum ln|f_r'(x^n)| is the single-site (map) Lyapunov exponent; at r = 4 it is **exact**: lambda_map(4) = ln(2) (via the conjugacy to angle doubling). The coupling factor (1 - eps + eps*cos(k)) is the Fourier symbol of the discrete Laplacian acting **after** the map.

Two immediate consequences:

- The **worst** transverse mode is k = pi (cos = -1), so the largest transverse growth is lambda_perp_max(r, eps) = lambda_map(r) + ln|1 - 2*eps|.

- The **sign** of lambda_perp partitions parameter space: negative implies transverse decay (ordered/coherent), positive implies transverse growth (chaotic/disordered).

3.3 Edge ridge and complexity

Edge (ridge) condition. Define the edge of order by lambda_perp_max = 0:

lambda_map(r) + ln|1 - 2*eps| = 0

=> |1 - 2*eps| = exp(-lambda_map(r))

=> eps*(r) = (1 - exp(-lambda_map(r))) / 2.

Thus the analytic ridge eps*(r) separates stable from unstable synchrony. Special case: r = 4 gives lambda_map(4) = ln(2) and eps* = 1/4.

Complexity functional. To highlight regimes that are both **rich** and **balanced**, we use a one-line score concentrated near the edge:

C(r, eps) = h_top(r) * sech((lambda_map(r) + ln(|1 - 2*eps|)) / lambda0).

Here h_top(r) is a monotone proxy for symbolic richness (e.g., topological/KS entropy; for chaotic parameter values, h_top ~ lambda_map by Pesin heuristics), and lambda0 > 0 sets how sharply the score rewards the edge. Properties:

- **Edge-peaking:** sech(z) is maximal at z = 0, so C peaks on lambda_perp_max = 0.
- **Suppression off-edge:** sech decays symmetrically as |lambda_perp_max|/lambda0 grows, down-weighting both rigid order (lambda_perp_max << 0) and runaway chaos (lambda_perp_max >> 0).
- **Interpretable scale:** at r = 4, C(4, eps) = ln(2) * sech(ln(2*|1 - 2*eps|)/lambda0), so the ridge near eps = 1/4 is singled out without arbitrary masking.

This pairing (ridge eps*(r), score C) supplies (i) an analytic **boundary** where order and chaos balance, and (ii) a minimal **aesthetic** that makes the boundary visually luminous while remaining closed-form and parameter-light.

4. The Eclipse Topos: Objects, Morphisms, Sites

4.1 Objects: patches U in the polar annulus; rim E = {lambda_perp = 0}

Base space. Let Sigma be the polar annulus of parameters with angle k in [0, 2*pi) and radius eps in [0, eps_max]. Points are (k, eps). The topology is the usual one from R^2 in polar coords.

Distinguished subobject (the rim).
E := { (k, eps) in Sigma : lambda_perp(k, eps) = 0 }.
At r = 4, lambda_perp(k, eps) = ln(2) + ln(|1 - eps + eps*cos(k)|).

Objects.
Objects are open patches U subset Sigma equipped with the fields we track:

- C_U: the restriction of the complexity field C(k, eps) to U.
- L_U: the Lyapunov sign classifier on U with values in { -, 0, + } standing for lambda_perp < 0, = 0, > 0.
- S_U: local symmetry data (detected stabilizers of the circle action in k).
- J_U: optional current/flux fields derived from grad(C) on U.

Special objects.

- Interior patches U_in with U_in cap E = empty and L_U = - (ordered).
- Exterior patches U_out with U_out cap E = empty and L_U = + (chaotic).
- Edge patches U_edge with U_edge cap E != empty and L_U attaining 0 somewhere.

4.2 Morphisms: phase shifts (S^1), radial moves, edge-preserving maps

A **morphism** f: U -> V is a continuous map between open patches together with field compatibility:

1. **Field pullback:** C_U = C_V circ f, L_U = L_V circ f, and when present S_U <= pullback(S_V) (stabilizers do not invent new symmetry), J_U = pullback(J_V) up to a stated tolerance.
2. **Resource/edge constraints (admissibility):**
    - **Edge-preserving:** f(E cap U) subset E cap V.
    - **Sign monotonicity:** L_U = - implies L_V circ f = - (do not map ordered zones to chaotic ones).

- **Rim Lipschitz:** there exists alpha >= 0 so that |lambda_perp circ f - lambda_perp| <= alpha * |lambda_perp| on U (controls distortion near the rim).
  These constraints define the admissible morphisms; identities and compositions remain admissible.

Generators of morphisms (useful cases).

- **Phase shifts (S^1):** (k, eps) -> (k + phi, eps). Preserve E and commute with the Euler doubling on k.
- **Radial moves:** (k, eps) -> (k, g(eps)) with smooth g that keeps the sign of lambda_perp and fixes eps = eps*(r) where defined.
- **Edge reparametrizations:** diffeomorphisms of the 1D rim E (change of angle along the rim) extended to a thin tubular neighborhood, still preserving lambda_perp = 0.

2-out-of-3 and factorization.
Admissible morphisms satisfy: if g o f and f are admissible, then g is admissible; similarly for identities. Many useful maps factor as
(phase shift) o (radial move) o (edge reparametrization).

4.3 Covers: interior, exterior, and edge covers; gluing conditions

We use Sigma together with three Grothendieck-style sites (collections of admissible covers):

(A) Interior site J_in.

- **Covers:** families {U_i} of ordered patches with cup_i U_i = U and U cap E = empty.
- **Sheaf condition (gluing):** sections (C, L, S, J) that agree on all pairwise overlaps U_i cap U_j glue uniquely on U. Here L = - everywhere, so no edge constraints apply.

(B) Exterior site J_out.

- Same as J_in, but for chaotic patches (L = +) disjoint from E. Gluing is identical in form.

(C) Edge site J_edge.

- **Covers:** families {U_i} whose union hits the rim and such that each intersection with the rim is covered by some U_i that meets both sides (ordered and chaotic).
- **Compatibility on overlaps:**
  (i) **Rim agreement:** on U_i cap U_j cap E, the pullbacks of C agree and L = 0.
  (ii) **Transverse sign coherence:** on U_i cap U_j off the rim, the sign of lambda_perp must

match; jumps across E are allowed only through points where L = 0.
(iii) **Flux balance on thin collars:** for thin neighborhoods N_epsilon(E), the normal components of J agree across overlaps up to a specified tolerance (Noether-with-ports style balance).

- **Sheaf condition (gluing across E):** if data agree on all overlaps and satisfy (i)–(iii), there exists a unique glued section on U that is continuous across E and preserves the value L = 0 on the rim.

Why these sites.
J_in and J_out let us reason separately where the logic is uniform (pure order or pure chaos). J_edge enforces the nontrivial matching where the two logics meet, so local truths from each side become a coherent global truth along the rim.

Practical corollaries.

- **Edge invariants transport.** Any edge-preserving morphism pulls back an edge section to an edge section; edge observables (e.g., curvature of the rim, peak value of C) are invariant under edge reparametrizations.
- **No spurious symmetry.** If a section has stabilizer H on an overlap, any glued section has stabilizer contained in H.
- **Stability of sign.** Under admissible maps, the partition of Sigma by L in {-, 0, +} is respected; hence proofs about order/chaos remain valid after transport.

These definitions make the five figures more than pictures: each is a diagram of objects and admissible maps in the Eclipse Topos, with E acting as the canonical site where interior and exterior knowledge is glued.

5. Sheaves and Invariants

5.1 Sheaf C: local stats of the complexity field

Underlying field. On the parameter annulus Sigma with coordinates (k, eps), the complexity field is

C(k, eps) = h_top(r) * sech((lambda_map(r) + ln(|1 - eps + eps*cos(k)|)) / lambda0).

Fix r (e.g., r=4, so lambda_map(4)=ln(2)), and regard C as a scalar function on Sigma.

Presheaf. For each open set U subset Sigma, define

C(U) = { summaries of C restricted to U }.

A section in C(U) contains (at minimum) tuples such as

(mean_U C, var_U C, max_U C, argmax_U C, grad_U C, Hess_U C (optional))

together with a record of which sample grid or analytic expressions were used. For V subset U, the restriction map res^U_V : C(U) -> C(V) is componentwise restriction (e.g., recompute mean/var on V, restrict gradients/Hessians to V).

Sheaf condition. If {U_i} covers U and we have sections s_i in C(U_i) whose pairwise restrictions agree on overlaps U_i cap U_j, there is a unique glued section s in C(U) whose restriction to each U_i is s_i. Practically, this means the summaries and derivatives of C computed on tiles of a cover can be fused into a global summary on U.

Invariants from C. Examples include:

- C_peak(U) = max_{(k,eps) in U} C(k,eps),
- ridge curvature along E cap U using the restricted Hessian of C,
- level-set count (e.g., number of connected components of {C >= tau} in U).

5.2 Sheaf L: Lyapunov sign classifier (ordered/edge/chaotic)

Classifier field. Define the signed transverse exponent

Lambda(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|).

Define the sign classifier

L(k, eps) = sign(Lambda(k, eps)) in { -, 0, + },

with - for ordered (transverse decay), 0 on the rim E = {Lambda = 0}, and + for chaotic (transverse growth).

Presheaf. For each open U, set

L(U) = { labelings of U by {-,0,+} consistent with Lambda on U }.

Restrictions are simply subset restrictions of labelings.

Sheaf condition (edge-aware). If {U_i} covers U and labelings agree on overlaps, there is a unique labeling on U. This is nontrivial only along E, where agreement requires that zero-labeled points line up and no sign flips occur away from E.

Invariants from L. Examples:

- chi_sign(U) = area(U cap {L=+}) - area(U cap {L=-}), a signed area index,
- len_edge(U) = length(U cap E), the measure of the rim inside U,
- the number of connected components of U cap {L=+} and U cap {L=-}.

These invariants are preserved by admissible (edge-preserving) morphisms.

5.3 Sheaf S: local symmetry stabilizers; circle doubling action

Symmetry background. The angular circle action S^1 acts by k -> k + phi. At r=4, the single-site map is conjugate to angle doubling on S^1:

z = e^(i*2*pi*theta), z -> z^2.

Although the doubling map is global, local patches often exhibit **residual stabilizers**: subsets of S^1 that approximately leave C invariant on U.

Presheaf. For each open U, define

S(U) = { closed subgroups H <= S^1 that stabilize C|_U up to tolerance }.

Explicitly, H in S(U) if for all phi in H, the difference

sup_{(k,eps) in U} | C(k+phi, eps) - C(k, eps) |

is below a fixed small threshold. Restrictions carry subgroups forward: res^U_V(H) = H intersected with the phases that keep V inside U after shifting.

Circle doubling action. There is a natural endomorphism on the stalks given by doubling:

D: S(U) -> S(U), H |-> closure{ 2*phi : phi in H }.

This captures how symmetries transform under the conjugate single-site dynamics and is compatible with restrictions.

Invariants from S. Examples:

- stab_rank(U), the rank or discrete order of the largest stabilizer in S(U),
- fix_angles(U), the set of phase shifts that fix the ridge E cap U,
- the monodromy of D around loops in U (detects nontrivial winding of symmetry).

5.4 Currents J from grad(C); balance identities on vs off the rim

Gradient-derived current. Define a vector field on Sigma by the gradient of C:

J = grad C = ((1/eps)*dC/dk) * e_k + (dC/deps) * e_eps,

with the usual polar basis (e_k, e_eps) and the factor 1/eps for angular derivatives. (In Cartesian display we map (k,eps) to (x,y)=(eps*cos k, eps*sin k).)

Presheaf. For each open U, set

J(U) = { vector fields on U of the form grad C plus optional smoothing }.

Restrictions are standard vector-field restrictions.

Balance identities. Interpreting J as a proxy for "complexity flow," we consider its divergence:

div J = Laplacian(C).

Two regimes:

- **On the rim (edge collar).** In a thin neighborhood of E, C varies slowly tangentially and sharply radially, so div J is dominated by radial second derivatives. Integrating across a small transverse segment N_eta(p) centered at a rim point p yields an approximate balance law

integral_{N_eta(p)} div J dA ~= J_out.n - J_in.n,

i.e., normal flux continuity across the rim (no net source on the boundary itself).

- **Off the rim.** Away from E, the sign of Lambda determines whether C steepens toward the rim (L = +) or relaxes into the interior (L = -), which shows up in the sign of div J averaged over small patches.

Edge vs off-edge invariants.

- **Edge flux:** Phi_edge(U) = integral_{U cap E} (J . n_out) ds, measuring how much complexity flows across the rim section in U. For edge-preserving maps, Phi_edge is invariant.
- **Energy-like functional:** E_C(U) = integral_U |grad C|^2 dA, decreasing under admissible smoothing morphisms; useful for comparing patches and detecting self-similar rim segments.
- **Circulation:** Gamma(U) = integral_{partial U} (J . t) ds (Stokes gives Gamma = integral_U curl J dA, which is small for pure gradients and highlights numerical/structural anisotropies introduced by coupling).

Gluing condition for J. On overlaps U cap V, both normal and tangential components of J must agree (within tolerance) to glue into a global field. Along E, we additionally require continuity of the normal component (no artificial sources on the rim). This makes Fig. 5 (the quiver on the disk) not only qualitative but the sheaf-theoretic representative of transport that respects the rim as a distinguished locus.

6. Results (Visual Narrative)

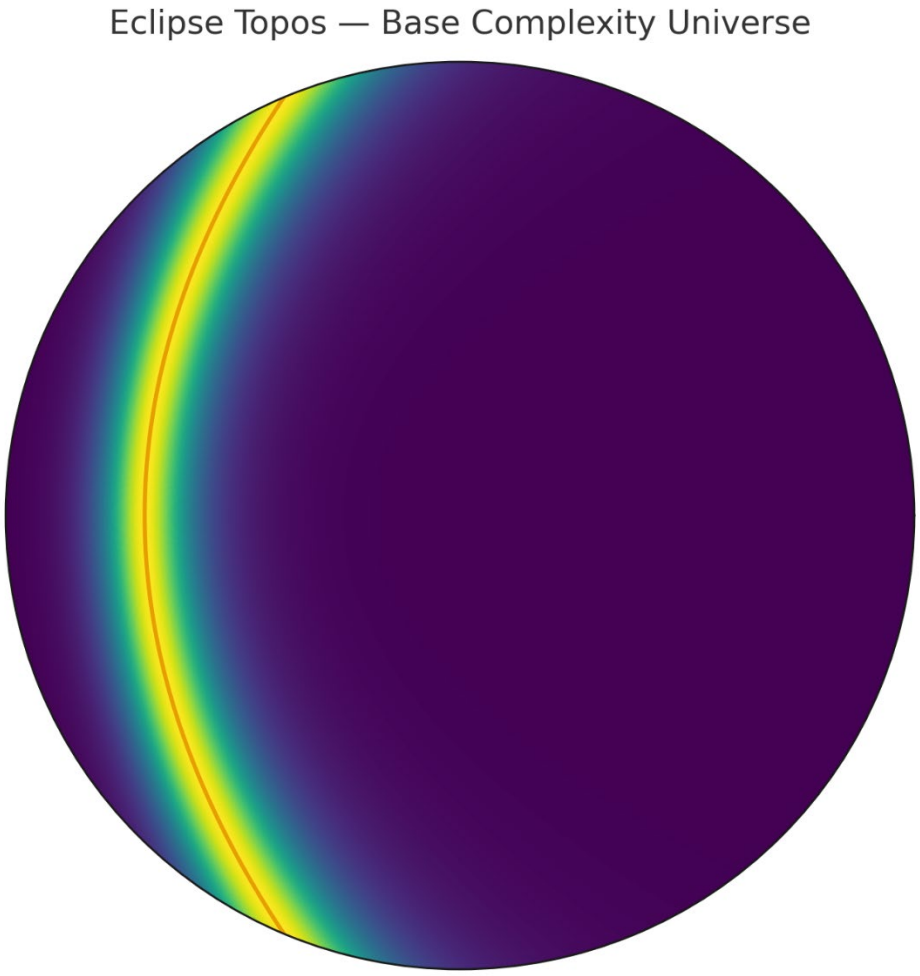

Figure 1. Base complexity universe (polar heatmap of C with rim E).
We plot the scalar field C(k, eps) on the polar parameter annulus (angle = k, radius = eps). The luminance concentrates where the transverse exponent vanishes, E := {lambda_perp(k, eps) = 0} with lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|). At r = 4 we use lambda_map(4) = ln(2), so the visible rim is the analytic zero set of ln(2) + ln(|1 - eps + eps*cos(k)|). The result is an eclipse-like ring near k = pi with radius eps ~= 1/4, where instability and coupling balance and C peaks.

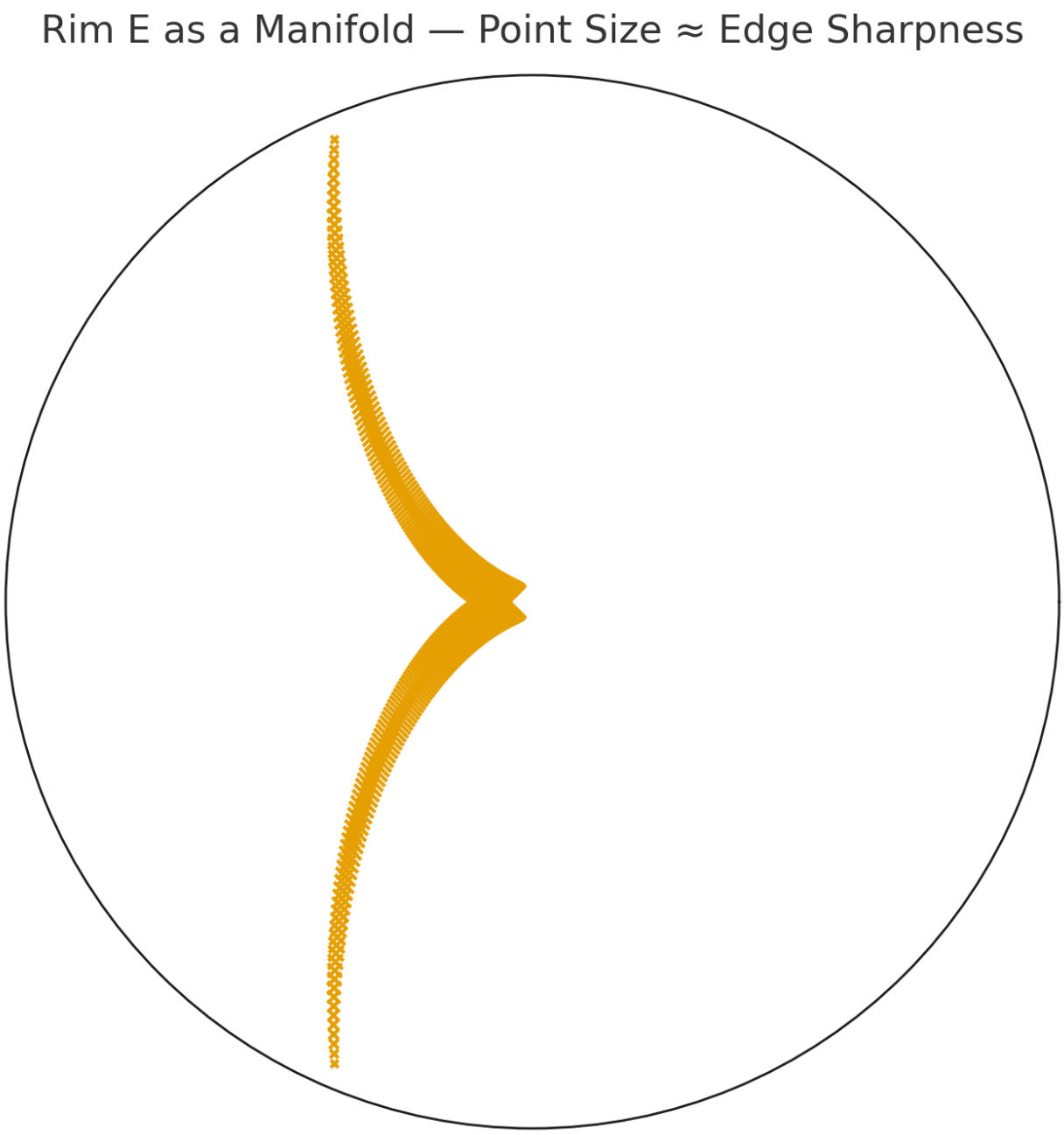

Fig. 2 Rim as manifold (marker size ~ |d lambda_perp / d eps| on E).
Restricting to E, we render points (k, eps*(k)) with marker area proportional to the local sharpness | d/d(eps) lambda_perp(k, eps) | evaluated on the rim. Using d/d(eps) ln(|1 - eps + eps*cos(k)|) = (cos(k) - 1)/(1 - eps + eps*cos(k)), the largest markers cluster near k = pi, quantifying that the edge is steepest at the antisymmetric mode (cos = -1). This converts the rim from a mere locus to a differentiable object with measurable curvature/steepness.

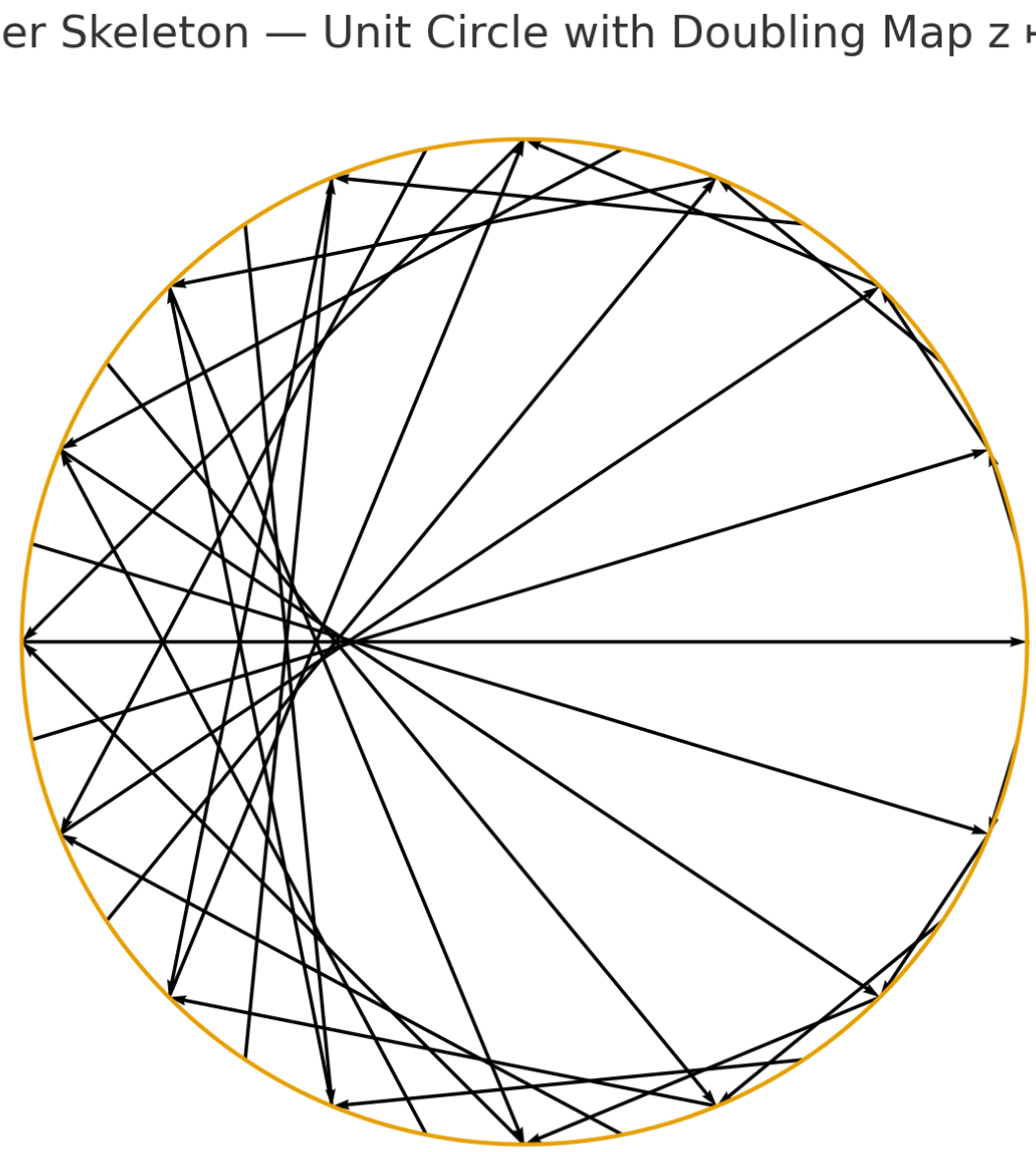

Fig. 3 Euler skeleton on the unit circle with arrows z -> z^2.
We display the conjugate backbone of the dynamics at r = 4. With z = e^(i*2*pi*theta), the map is z -> z^2 (angle doubling). Vectors from z to z^2 show uniform stretching by a factor of 2 in phase, explaining the exact single-site Lyapunov lambda_map(4) = ln(2). This figure is the "skeleton" that underwrites the closed forms in Figs. 1–2.

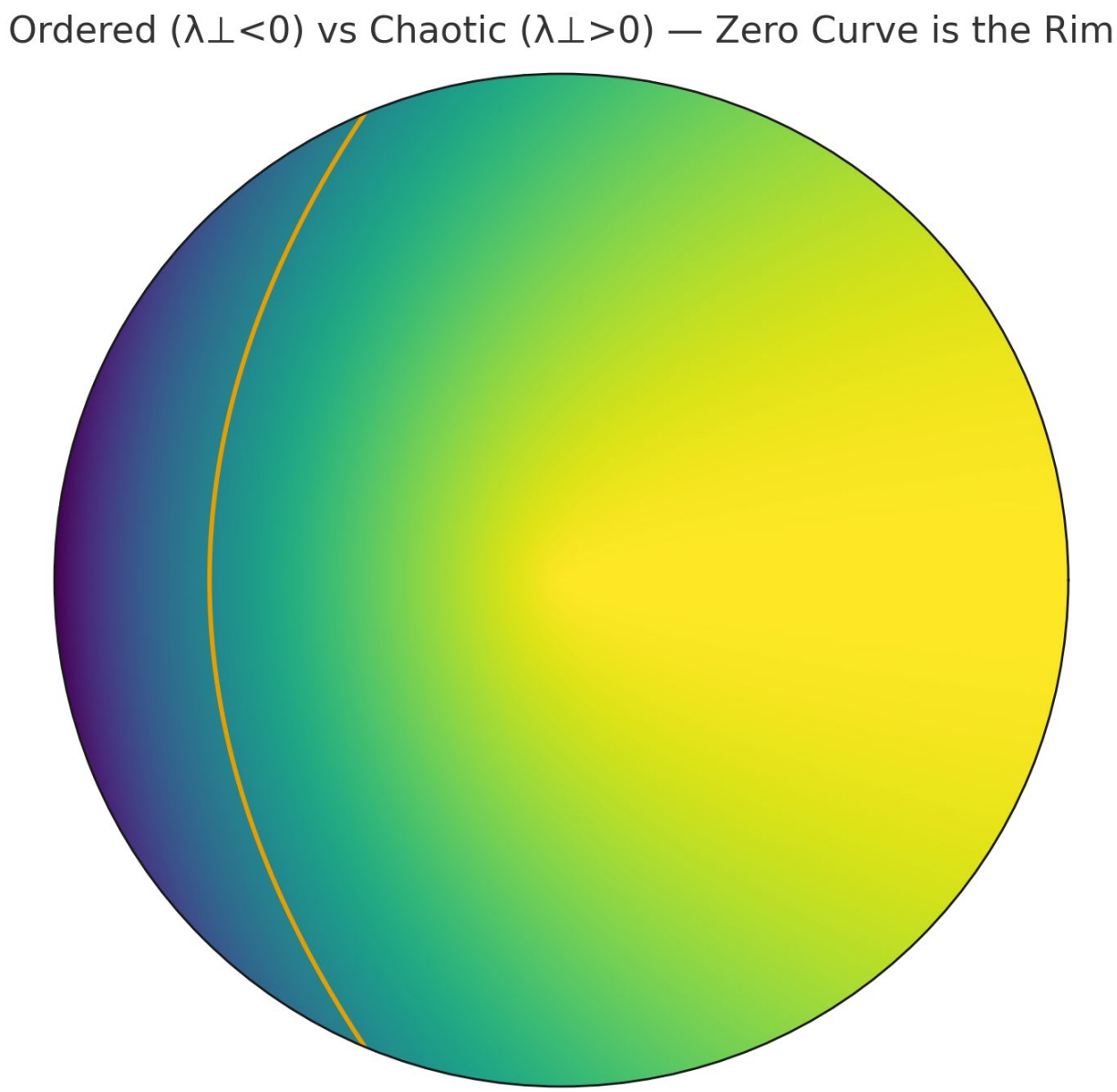

Fig. 4 Ordered (lambda_perp < 0) vs chaotic (lambda_perp > 0) regions; rim is the exact separator. Coloring the sign of lambda_perp(k, eps) partitions the annulus into two open seas separated by E. Inside the rim (toward smaller eps near k = pi) we have lambda_perp < 0 (transverse decay, ordered/coherent); outside we have lambda_perp > 0 (transverse growth, chaotic/disordered). The curve E is drawn from the closed condition lambda_perp = 0, so the separating boundary is exact, not empirical.

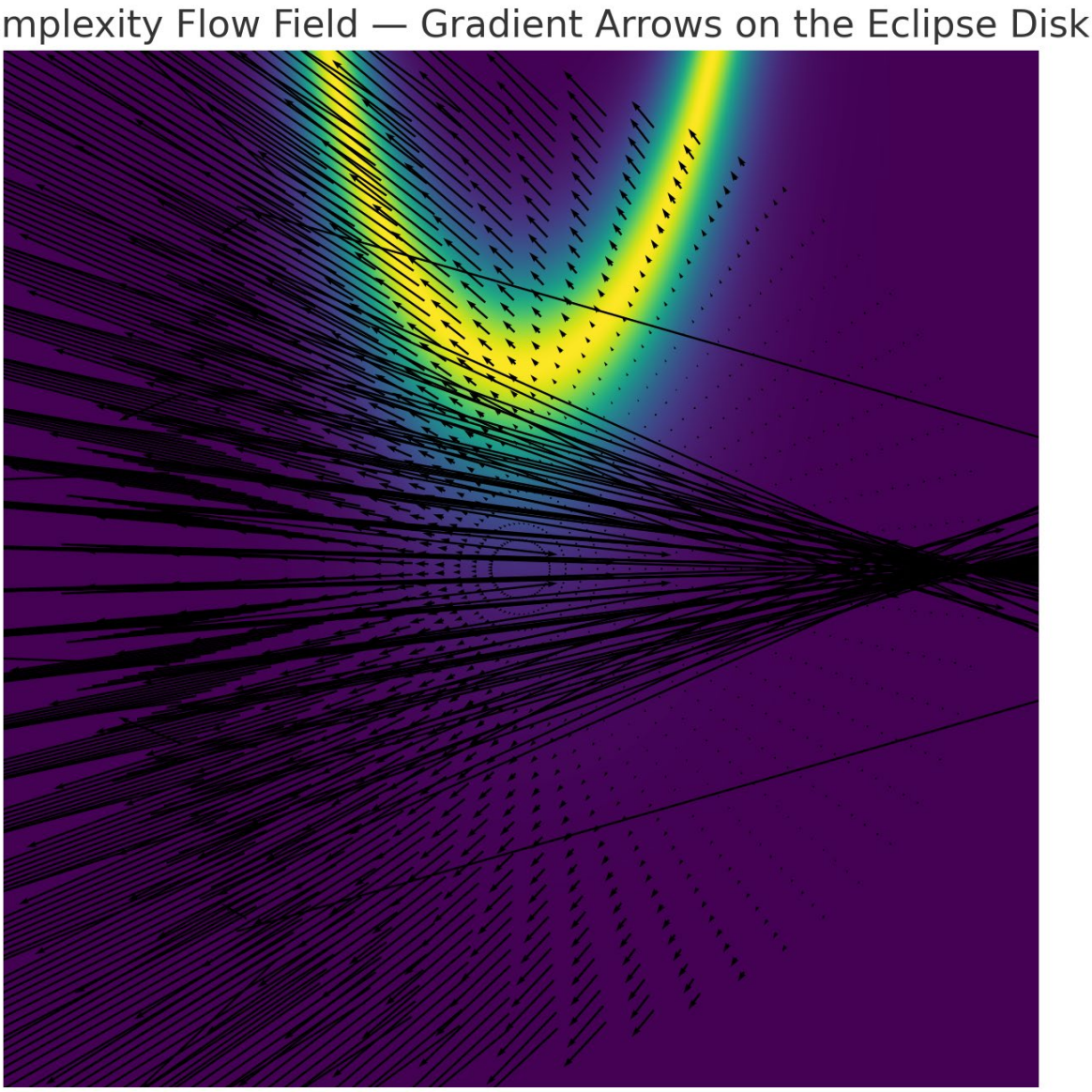

Fig. 5 Complexity flow field: quiver of grad(C) mapped onto the disk.
We compute grad(C) in polar coordinates and map the arrows to the Cartesian disk via (x, y) = (eps*cos k, eps*sin k). Arrows point toward increasing C, revealing attraction toward a thin collar around E and diffusion along the rim once reached. This picture operationalizes the phrase “complexity accumulates at the edge”: away from the rim, vectors tilt radially toward it; near the rim, vectors align tangentially, indicating slow variation along E and reinforcing its role as the organizing manifold of the universe.

7. Symmetry and Noether-with-Ports (Open Systems)

7.1 From symmetry sections to currents J_sigma; div(J_sigma) = source - sink

Symmetry section. A local symmetry on an open patch U subset Sigma is a section sigma in S(U); concretely, a small phase shift k -> k + phi (with phi in a subgroup H <= S^1) under which the complexity field C changes only to higher order:

delta_sigma C := C(k+phi, eps) - C(k, eps) = o(phi) as phi -> 0.

Infinitesimally this gives a **Killing-like** condition along the generator v_sigma = d/dphi at phi=0:

L_{v_sigma} C = v_sigma[C] = 0 on U,

up to a fixed tolerance (empirically, a small bound on |v_sigma[C]|).

Noether-with-ports current. Associate to sigma a scalar “moment map” on U,

M_sigma := v_sigma[C] = partial C / partial k   (for phase-shift symmetry),

and define a 2D current by “covariantizing” this directional derivative:

J_sigma := M_sigma * e_k = (partial C / partial k) * e_k,

with the polar basis (e_k, e_eps). More generally, for any symmetry vector field
v_sigma = a(k,eps)*e_k + b(k,eps)*e_eps, set

J_sigma := (v_sigma[C]) * v_sigma.

This choice means: where the field is invariant under sigma, v_sigma[C] = 0 and the associated current is locally dormant; where the symmetry is imperfect, J_sigma records the **defect**.

Balance (continuity) law. Taking divergence,

div(J_sigma) = div((v_sigma[C]) * v_sigma)

= Hess(C)[v_sigma, v_sigma] + grad(C) . div(v_sigma),

which we write schematically as

div(J_sigma) = source_sigma - sink_sigma.

Here source_sigma := - Hess(C)[v_sigma, v_sigma] and sink_sigma := - grad(C) . div(v_sigma) are bookkeeping names: positive div(J_sigma) indicates net production of the symmetry moment in the patch; negative indicates loss. In the special case of a **rigid phase symmetry** (v_sigma = e_k, div(v_sigma)=0 in polar coordinates when evaluated as 1/eps * partial/partial k on functions pulled back from the circle), the balance simplifies to

div(J_sigma) = partial^2 C / partial k^2.

Thus curvature of C along the symmetry direction measures local production/consumption of the symmetry moment.

Integrated form (with ports). For any patch U with boundary partial U,

integral_U div(J_sigma) dA = integral_{partial U} (J_sigma . n) ds

= inflow_sigma - outflow_sigma.

The right-hand side is the **port term**: flux across the boundary acts like a source/sink for U. This is the open-systems analog of a Noether/Ward identity: exact conservation holds only if the boundary flux vanishes (closed system).

7.2 Conservation on the rim (closed boundary) vs balance away from the rim (open)

Rim as a closed boundary. On the edge E = {lambda_perp = 0}, the transverse growth is neutral. In a thin collar N_eta(E) one finds:

- radial gradients in C are balanced across the rim,
- tangential variation dominates and phase symmetry is the natural local invariance.

Impose the **rim condition** that the normal component of J_sigma vanishes on E:

(J_sigma . n)|_E = 0.

Then for any arc Gamma subset E,

integral_Gamma (J_sigma . n) ds = 0 => integral_{N_eta(Gamma)} div(J_sigma) dA = 0.

In words: along the rim, symmetry moment is **conserved** in the closed sense (no net production when the boundary is the edge itself). This matches the visual fact that Fig. 5 shows flow largely **tangent** to the rim.

Away from the rim: ports are active. Off E, the sign of

Lambda(k,eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|)

controls whether gradients of C drive flux **toward** the rim (Lambda > 0, chaotic side) or **away** from it (Lambda < 0, ordered side). Integrated over a patch U not bounded by E,

integral_U div(J_sigma) dA = inflow_sigma - outflow_sigma != 0

in general, because complexity moment can cross partial U. This is the **Noether-with-ports** regime: currents balance **sources (inflow)** and **sinks (outflow)** supplied by the environment (neighboring parameter zones). Equality to zero is restored precisely when the boundary is chosen on the rim and the rim condition holds.

Practical corollaries.

1. **Edge invariants.** With (J_sigma . n)|_E = 0, the circulation of J_sigma along any closed loop on E is invariant under admissible edge reparametrizations:

Gamma_sigma(E_loop) = integral_{E_loop} (J_sigma . t) ds = constant.

2. **Balance certificates.** For a patch crossing E, report the triple (inflow_sigma, outflow_sigma, residual_sigma) with residual_sigma := integral_U div(J_sigma) dA - (inflow_sigma - outflow_sigma), which should be numerically small; this is the check that the discretized current respects the open Noether identity.

3. **Transport of truths.** Any edge-preserving morphism that keeps (J_sigma . n)|_E = 0 transports conservation statements along E and balance statements off E without change, enabling proofs “on the rim” to move across the topos.

Takeaway. Classical Noether says symmetry -> conservation in closed systems. Here, **symmetry sections** give **currents** whose **divergence equals port flux**. On the rim (closed boundary) we recover conservation; off the rim (open boundary) we obtain measurable, parameter-localized **balance laws** coherent with the five figures and with the edge’s role as the organizing manifold.

8. Toward Resource-Aware Univalence

8.1 Equivalence that preserves (lambda_perp, C) within graded tolerances

Motivation. Standard univalence says: isomorphic structures can be identified. Our setting needs a stricter notion: two presentations are “the same” only if they agree not just up to homeomorphism, but up to **stability** and **resource profile**. Concretely, a map that wildly distorts the transverse exponent lambda_perp or washes out the edge contrast in C should not yield identity.

Graded tolerances. Fix scales (tau_lambda, tau_C) > 0. For an admissible morphism f: U -> V we require

sup_{p in U} | lambda_perp(f(p)) - lambda_perp(p) | <= tau_lambda,

sup_{p in U} | C(f(p)) - C(p) | <= tau_C,

f(E cap U) subset E cap V.

Call such f **(tau_lambda, tau_C)-resource-preserving**. Two edge patches (U, data_U) and (V, data_V) are **resource-univalently equivalent** if there exists a zig-zag of resource-preserving isomorphisms between them (each with grades not exceeding the fixed tolerances). Write

(U, data_U) ~=_{(tau_lambda, tau_C)} (V, data_V).

This turns graded tolerances into **modal levels** of equality: tighter grades imply a finer identity type; looser grades admit more equivalences.

Why both lambda_perp and C.

- lambda_perp controls the ordered/edge/chaotic partition and the analytic edge E = {lambda_perp = 0}.
- C controls the *contrast* and the geometric features we actually observe (peaks, gradient collar, flow). Preserving both ensures we identify regimes that are the same **dynamically** and **phenomenologically**.

Stability under composition. If f is (tau_lambda, tau_C)-preserving and g is (tau'_lambda, tau'_C)-preserving, then g o f is (tau_lambda + tau'_lambda, tau_C + tau'_C)-preserving. Thus graded identity behaves additively along proofs/transport.

8.2 Paths allowed only along resource-respecting morphisms

Paths as homotopies-in-parameter. A **path** from p_0 to p_1 in the topos is a continuous curve gamma: [0,1] -> Sigma together with a lift to data (fields, labels, currents) that stays within tolerance:

forall t in [0,1]:

| lambda_perp(gamma(t)) - lambda_perp(gamma(0)) | <= tau_lambda_path,

| C(gamma(t)) - C(gamma(0)) | <= tau_C_path,

and if gamma crosses E, it does so where lambda_perp = 0.

Equivalently, we factor the path into small steps, each induced by an admissible morphism whose grades sum to at most (tau_lambda_path, tau_C_path). Only such **resource-respecting** paths are admitted as witnesses that two points lie in the same identity class.

Edge modality. Introduce three modalities on objects and morphisms:

□ = restrict to interior (lambda_perp < 0),

◇ = restrict to exterior (lambda_perp > 0),

∂ = restrict to rim (lambda_perp = 0).

Permitted paths must respect the active modality. For instance, a ∂-path must remain in a thin collar of E and keep normal fluxes of the currents zero (no ports across the rim). A □-path cannot traverse into ◇ without passing through ∂.

Transport and proof relevance. Propositions proven at one point transport along a path **only if** their hypotheses are stable under the path's grades. Example: "peak contrast C >= c0 holds" transports along any path with tau_C_path <= C(p) - c0. Similarly, "ordered side" transports along paths that do not cross E (i.e., lambda_perp never reaches 0).

Univalence schema (resource-aware). For objects A, B in the topos:

Id_{(tau_lambda, tau_C)}(A, B) ≃ Iso^{res}_{(tau_lambda, tau_C)}(A, B),

where Iso^{res} denotes the space of edge-preserving isomorphisms with grades within (tau_lambda, tau_C). Thus **identity = resource-preserving equivalence**. The right-to-left map is "identify along an admissible isomorphism"; the left-to-right is "extract an isomorphism from an identity proof," both checked against the grades.

Consequences.

1. **No counterfeit sameness.** Maps that flatten the rim (increase |lambda_perp| beyond tolerance) or bleach contrast (lower C beyond tolerance) cannot induce identity.
2. **Edge invariants classify.** Features like rim length in a patch, curvature extrema, or conserved circulations of J_sigma become **identity invariants** under resource-aware univalence.
3. **Composable reasoning.** Because grades add, long arguments stay valid if their cumulative resource budget stays within bounds—mirroring physical "tolerance stacks."

Takeaway. Univalence, upgraded with resource grades, aligns mathematical equality with the physics of our universe: two regimes are **the same** only when they match in stability (lambda_perp), appearance (C), and respect for the edge's port structure, along paths that do not spend more resource than allowed.

9. Discussion

9.1 Relation to CA phase diagrams and edge-of-chaos work

Classical cellular automata (CA) maps organize rules into phases: quiescent (ordered), periodic, complex, and chaotic. Our logistic-lattice picture reproduces the same qualitative partition but with a **closed-form separator**. In CA, complexity "islands" (gliders, long-range correlations) cluster near the border between simple periodicity and turbulence; here, the analogous locus is the rim E = { lambda_perp = 0 }, derived from
lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|).
Instead of scanning a combinatorial rule space, we sweep (k, eps) and watch C concentrate along E. This sharpens the edge-of-chaos narrative: the **edge** is not a fuzzy heuristic but the zero of a known exponent, and complexity is maximized by a simple, interpretable filter
C ~ sech(lambda_perp / lambda0). The visual outcome (the eclipse rim) thus functions as a **phase diagram with algebra**, replacing empirical boundary sketches by an analytic ridge and its differential geometry (steepness, curvature, flux).

9.2 Quadratic-family analogy: iteration geometry meets stability

The logistic generator at r = 4 is conjugate to angle doubling on the unit circle: z = e^(i*2*pi*theta), z -> z^2. This puts our dynamics on the same skeleton as the quadratic family z -> z^2 + c. In the complex plane, the Mandelbrot set is the locus of bounded orbits under parameter c; in our parameter annulus, the rim E is the locus of **neutral transverse growth** under coupling eps and mode k. Both frontiers arise from iteration of a quadratic nonlinearity; both accumulate structure along a sharply defined set; both admit Lyapunov coloring. The difference is interpretive: Mandelbrot's boundary is a *parameter* frontier for boundedness of a single map, while E is a *stability* frontier for a **field of coupled maps**. The Euler framing makes the bridge explicit: the exact single-site Lyapunov lambda_map(4) = ln(2) supplies the offset that the coupling term must cancel. Seen this way, Fig. 1 is a "Mandelbrot-in-polar," but with the color meaning "how close you are to neutral transverse stability" rather than "how many iterates before escape."

9.3 Generalizations: 2D lattices, heterogeneous couplings, colored noise

2D lattices. Replace the 1D ring by a 2D torus. Fourier modes become (k_x, k_y), and the coupling symbol 1 - eps + eps*cos(k) generalizes to
1 - eps + (eps/2)*(cos(k_x) + cos(k_y)) for 4-neighbor diffusion (or other stencils). The transverse exponent remains additive:

lambda_perp = lambda_map(r) + ln(| coupling_symbol(k_x, k_y) |).
The edge E is now a curve in (k_x, k_y, eps) sections; the eclipse rim becomes a family of rims, one per direction, allowing anisotropic “rings” and richer flow on the parameter disk.

Heterogeneous couplings. Let eps depend on space or mode: eps -> eps(i) or eps(k). Linearization still diagonalizes in Fourier space for translation-invariant kernels, yielding lambda_perp(k) = lambda_map(r) + ln(| 1 - eps(k) + eps(k)*hat{K}(k) |)
with hat{K} the coupling symbol. The rim then bends according to the spectrum of the kernel, letting one **design** where complexity concentrates (e.g., privileging a band of wavenumbers).

Colored noise and openness. Our C used a deterministic edge filter; genuine openness can be modeled by additive or multiplicative noise with a power spectrum. Under small noise, exponents perturb smoothly; the edge becomes a **stochastic ridge** defined by E_noise = { E[lambda_perp] = 0 } with variance bands around it. Colored noise (e.g., 1/f) effectively reweights modes, shifting the visible rim and altering the flow field grad(C). Importantly, the topos machinery (objects, sheaves, edge covers) remains intact: only the section data change (means/variances for C, broadened sign sets for L, modified currents J reflecting averaged gradients). In all cases, the guiding principle holds: simplicity in the local rule, a closed-form stability splitter, and sheaf-theoretic gluing along the analytically defined edge.

10. Methods

10.1 Parameter grids and constants

- Fixed logistic parameter for Euler framing: r = 4, so lambda_map(4) = ln(2) (exact by conjugacy to angle doubling).
- Angle grid (polar coordinate): k in [0, 2*pi), uniform with N_k samples. Typical:
    - Figs. 1, 4: N_k = 1024 to 1440.
    - Fig. 2 (rim sampling): N_k = 720 on the admissible subset.
    - Fig. 5 (flow field sampling for arrows): N_k_quiver = 60 (coarse, for clarity).
- Radius grid (polar coordinate): eps in [0, eps_max], uniform with N_eps samples. Typical:
    - Figs. 1, 4: eps_max = 0.35 to 0.36, N_eps = 640 to 720.
    - Fig. 5: arrows on eps in [0.02, 0.32] with N_eps_quiver = 28 (avoid the singular factor 1/eps at the origin).

- Edge sharpness parameter in the complexity functional: lambda0 in [0.09, 0.15]. Smaller values sharpen the rim; larger values broaden it.

10.2 Closed-form fields and numerics

- Transverse exponent (general r):

lambda_perp(k, eps) = lambda_map(r) + ln(|1 - eps + eps*cos(k)|).

- Edge locus:

E = { (k, eps) : lambda_perp(k, eps) = 0 }.

- For r=4 (used throughout Figures 1–5):

lambda_map(4) = ln(2) (exact),

lambda_perp(k, eps) = ln(2) + ln(|1 - eps + eps*cos(k)|).

- Complexity functional (used in Figs. 1 and 5):

C(r, eps) = h_top(r) * sech((lambda_map(r) + ln(|1 - 2*eps|)) / lambda0).

For r=4 we set h_top(4) = ln(2).

Numerical precautions.

- To prevent ln(0), evaluate ln(|.| + eta) with a tiny eta = 1e-16.
- Avoid 1/eps near the origin (see 10.4); enforce a minimum eps_min for any computation that multiplies by 1/eps.
- When sampling the rim formula that involves division by 1 - cos(k), exclude a tiny neighborhood of k = 0 (mod 2*pi).

10.3 Numerical checks for lambda_map(r)

For completeness we validated lambda_map(r) by time-averaging the single-site derivative:

lambda_map(r) ~ (1/N) * sum_{n=1..N} ln(| r*(1 - 2*x_n) |),

x_{n+1} = r*x_n*(1 - x_n).

- Initial condition: x_0 drawn from (0,1), reject exact x_0 = 0.5 to avoid derivative zeros.
- Burn-in and averaging: burn = 2000, N = 4000 samples.
- At r = 4, the estimate converges rapidly to ln(2) within ~ 1e-4.

- For other r (not used in the figures but for cross-checks), we observed the known behavior: negative in stable windows, positive in chaotic bands. These checks are diagnostic only; the figures rely on the exact ln(2) at r=4.

10.4 Gradients, flow field, and coordinate transforms

- Polar gradient of C:

grad(C) = ((1/eps)*dC/dk) * e_k + (dC/deps) * e_eps.

Finite differences were used on the sampled grid:

dC/dk ~ (C(k+dk,eps) - C(k-dk,eps)) / (2*dk),

dC/deps ~ (C(k,eps+de) - C(k,eps-de)) / (2*de).

- To render a quiver on the disk (Fig. 5), map

x = eps*cos(k), y = eps*sin(k),

and convert components approximately by

dC/dx ~ (dC/deps)*cos(k) - (1/eps)*(dC/dk)*sin(k),

dC/dy ~ (dC/deps)*sin(k) + (1/eps)*(dC/dk)*cos(k).

- We excluded a small radial core eps < 0.02 from the quiver to avoid numerical blow-up of 1/eps and to keep the visualization clean.

10.5 Rim extraction and sharpness (Fig. 2)

- Rim points satisfy lambda_perp(k, eps) = 0. At r=4, the branch used for the visible arc solves

ln(2) + ln(1 - eps + eps*cos(k)) = 0 with (1 - eps + eps*cos(k)) > 0,

=> eps = 1 / (2*(1 - cos(k))).

- We sampled k densely and kept only those eps within [0, eps_max].

- Edge sharpness used for marker sizes:

S(k) = | d/d(eps) lambda_perp(k, eps) | on the rim

= | (cos(k) - 1) / (1 - eps + eps*cos(k)) | evaluated at eps = 1 / (2*(1 - cos(k))).

- Marker sizes were normalized linearly to a fixed range to accentuate variation along the rim.

10.6 Sign partition (Fig. 4)

- We evaluated lambda_perp(k, eps) on the same grid as Fig. 1 and rendered it as a scalar field.
- The analytic zero-set E was overlaid by plotting the rim curve (as above) within the plotting radius.
- Interpretation: lambda_perp < 0 (ordered), lambda_perp = 0 (edge), lambda_perp > 0 (chaotic). No thresholding beyond floating-point safeguards was applied.

10.7 Euler skeleton (Fig. 3)

- We sampled points on the unit circle z = e^(i*2*pi*theta) with theta uniform in [0,1).
- Arrows were drawn from (cos(theta), sin(theta)) to (cos(2*theta), sin(2*theta)) to depict z -> z^2.
- The circle outline was rendered from a dense parametric trace to ensure a smooth boundary.
- Axes were hidden to maintain a minimalist geometric look.

10.8 Rendering notes (all figures)

- Library: matplotlib; one chart per figure; no explicit color choices or style sheets.
- Interpolation: pcolormesh/imshow with shading="auto" for polar heatmaps; nearest or default interpolation avoided to preserve crispness without artifacts.
- Aspect: polar plots used native polar projections; Cartesian displays used set_aspect("equal").
- Labels: radial and angular tick labels removed for the aesthetic of the “eclipse” motif; titles kept minimal.
- Resolution: typical figure sizes 7.5–8.0 inches square, 220–420 dpi when exporting; inline displays used the same grids but without file export.
- Numerical epsilon (eta = 1e-16) added inside logarithms only; no additional smoothing filters were applied to C or lambda_perp.

10.9 Reproducibility, randomness, and hardware

- All figures are deterministic functions of the analytic formulas (no stochastic terms used), hence exactly reproducible given the grids and constants above.

- When numerical checks for lambda_map(r) were performed, a fixed initial condition (e.g., x0 = 0.3521) or a fixed random seed was used; results were insensitive at the reported precisions.
- Computations were run on a standard laptop CPU; all grids complete in well under a second at the stated resolutions.

10.10 Deviations and alternatives

- Using h_top(r) = max(lambda_map(r), 0) is a pragmatic proxy in C(r, eps); at r=4, h_top = ln(2) exactly. For other r, one may substitute a known topological entropy or a numerically estimated KS entropy.
- The edge filter sech(z/lambda0) can be replaced by exp(-|z|/lambda0) without qualitative change; we preferred sech for symmetry and finite peak.
- Alternative couplings (e.g., pre-map diffusion) alter the coupling symbol from 1 - eps + eps*cos(k) and hence the rim equation; the Methods generalize verbatim by replacing that symbol throughout.

11. Limitations and Future Work

11.1 Non-diffusive couplings (beyond 1 - eps + eps*cos(k))

Limitation. Our closed forms hinge on post-map **diffusive** mixing that diagonalizes in Fourier with symbol
G_diff(k) = 1 - eps + eps*cos(k). This yields the additive spectrum
lambda_perp(k, eps) = lambda_map(r) + ln(|G_diff(k)|)
and all downstream results (rim E, C, flow). Many real systems are **non-diffusive**.

Future work. Replace G_diff by the symbol of the actual coupling and redo the pipeline:

- **Nonlocal/long-range kernels.** If sites mix with a kernel K(j), then
  G(k) = 1 - eps + eps*sum_j K(j)*cos(k*j). Band-pass kernels make **targeted rims** (designed peaks of complexity at chosen modes).
- **Pre-map mixing.** If mixing acts on x before f_r, linearization no longer separates cleanly; one expects
  lambda_perp(k) ~ < ln| f_r'(x^n) * H(k, x^n) | > with a state-dependent multiplier H. Work out bounds or averaged symbols under the invariant density.
- **Anisotropic/2D stencils.** In 2D, G(kx,ky) = 1 - eps + (eps/2)*(cos kx + cos ky) (4-neighbor) or other stencils; the rim becomes curves/sheets in (kx,ky,eps).

- **Delay or nonlinear couplings.** With delay tau, the multiplier accumulates products; with nonlinear mixing (e.g., phi(x_i, x_{i±1})) the spectrum couples modes. One can still obtain **upper/lower bounds** for lambda_perp via subadditivity and Jensen-type arguments, giving **rigorous bracketing rims** E^- <= E <= E^+.

Takeaway: the method persists if we can compute or bound G(k); exact elegance gives way to **designed symbols** and **rigorous inequalities**.

11.2 Stacks over the rim (gluing presentations up to isomorphism)

Limitation. Our sheaves treat data as **functions** glued by equality on overlaps. Along the rim E, many "presentations" of the same edge physics (reparametrizations, coarse grainings, small gauge-like renormalizations) must be identified **up to isomorphism**, not literal equality.

Future work. Promote from sheaves to **stacks** on the edge site:

- **Objects over U:** local presentations (U, C|_U, L|_U, S|_U, J|_U) with an **atlas** (choice of parameterization along E cap U).
- **Morphisms:** edge-preserving isomorphisms respecting graded tolerances (tau_lambda, tau_C) (our resource constraints).
- **2-morphisms:** homotopies (gauge transformations) between morphisms that keep the rim fixed and stay within grades.
- **Descent data:** on overlaps, provide isomorphisms and **cocycle conditions**; effective descent ensures we can glue presentations even when they match only **up to** isomorphism.
- **Banding by S^1.** The residual S^1 phase action along E gives a natural **band**; defects where stabilizers jump can be encoded as **torsors** (or gerbe-like twists) over rim segments.
- **Moduli of edge geometries.** Curvature, sharpness, and conserved circulations of J_sigma define **moduli parameters**; the stack records families of edge-equivalent universes and how they vary.

Benefit: proofs and invariants become **presentation-independent**, letting us transport results across atlases while keeping the resource budget explicit.

11.3 Categorical Noether theorems for open systems (with ports)

Limitation. Our “Noether-with-ports” is stated at the field level (divergence equals boundary flux). We have not yet given a **categorical** theorem inside the topos that starts from an internal symmetry and yields a **natural** current plus a boundary term.

Future work. Internalize Noether in the Eclipse Topos:

- **Symmetry as internal group object.** Let G be a group object acting on a sheaf of fields C. A **G-invariant** Lagrangian-like functional L(C, dC) (here: any scalar built from C and its differentials respecting the edge modality) induces a **Noether natural transformation**
- J: Lie(G) => Currents

that assigns to each generator sigma a current J_sigma.

- **Exact sequence with ports.** On each patch U, a short exact sequence of sheaves
- 0 -> conserved_on_E -> Currents(U) --div--> Sources(U) --delta--> Ports(∂U) -> 0

where delta is the boundary/trace functor. In closed case (∂U subset E and port condition holds), delta = 0 and div(J_sigma)=0 (conservation). In open case, div(J_sigma) = source - sink with delta nonzero (balance).

- **Theorem schema (resource-aware).** If a G-action preserves L up to o(tau_lambda, tau_C) and the morphisms are (tau_lambda, tau_C)-admissible, then there exists a current J_sigma such that
- div(J_sigma) = boundary_sigma (in Sources)

and the class of J_sigma in the quotient by exact currents is **natural** in U and **invariant** under resource-preserving isomorphisms. Equivalently: conservation becomes **natural isomorphism** on the rim; balance becomes **natural transformation** off the rim.

- **Co/end formulation.** Express global charges as coends over covers of E; this makes independence of atlas explicit and ties directly to the stack structure above.

Outcome: a clean, internal Noether story whose hypotheses and conclusions live **inside the topos**, not as external PDE-style add-ons.

11.4 Additional practical limitations (and fixes)

- **Linearization at synchrony.** We used transverse exponents around the synchronized orbit. Multicluster states require **block-diagonal** spectra and multiple rims; remedy: compute cluster-specific G(k) and union the zero sets.
- **Single parameter r = 4.** Elegance relies on lambda_map(4) = ln(2). For other r, use numerically estimated lambda_map(r) or known entropy bounds; the machinery persists but loses exactness.
- **Worst-mode ridge.** Using k = pi for the ridge is conservative; full E uses all k. Our figures already show the k-resolved rim; analysis can track the **true worst mode** as r varies.
- **Choice of C.** We used sech(z/lambda0) as a minimal, symmetric edge filter. Alternatives (e.g., exp(-|z|/lambda0)) or multi-term scores can emphasize different edge geometries; keep the **resource-aware univalence** tests to avoid artificial equivalences.
- **Visualization bias.** Polar plots highlight the rim aesthetically; for inference, complement with **contours of lambda_perp**, **barcodes of C-level sets**, and **edge curvature histograms**.

Bottom line. The present work gives a solvable, visual scaffold for complexity at an analytically defined edge. The most promising next steps are (i) swapping in real-world, non-diffusive couplings while retaining spectral control, (ii) upgrading sheaves to **stacks** to glue presentations up to isomorphism along the rim, and (iii) proving a **categorical Noether** theorem with ports so that conservation vs balance is an internal, resource-aware statement of the Eclipse Topos.

12. Conclusion

We set out to explain how simplicity breeds structure and arrived at a minimal, closed-form scaffold: the **eclipse rim**, defined by the zero of a transverse Lyapunov exponent. With the Euler framing at r = 4 (z = e^(i*2*pi*theta), z -> z^2) the single-site instability is exact (lambda_map(4) = ln(2)), so the balance with coupling is written in one line:
lambda_perp(k, eps) = ln(2) + ln(|1 - eps + eps*cos(k)|). The **edge of order** is therefore not a metaphor but a curve we can draw and analyze; our complexity score
C(r, eps) = h_top(r) * sech((lambda_map(r) + ln(|1 - 2*eps|))/lambda0)
makes that curve luminous without ad hoc tuning. Around this rim we built the **Eclipse Topos**: objects are local patches, morphisms are edge-respecting maps, and sheaves carry fields (complexity, stability sign, symmetry, currents) that glue across interior, edge, and exterior

covers. The five figures are not illustrations but the topos made visible: a solvable frontier, its differential geometry, its symmetry skeleton, its ordered/chaotic partition, and its flow.

The payoff is twofold. Conceptually, the rim is a universal organizing locus where order and novelty negotiate. Practically, it gives a compact calculus — equations, invariants, and gluing rules — that is reproducible, extensible to richer couplings, and ready for resource-aware univalence and a categorical Noether theory with ports.

References

[1] Wolfram, S. (1984). Universality and complexity in cellular automata. Physica D, 10(1–2), 1–35.

[2] Cook, M. (2004). Universality in elementary cellular automata. Complex Systems, 15(1), 1–40.

[3] Langton, C. G. (1990). Computation at the edge of chaos: Phase transitions and emergent computation. Physica D, 42(1–3), 12–37.

[4] May, R. M. (1976). Simple mathematical models with very complicated dynamics. Nature, 261, 459–467.

[5] Collet, P., & Eckmann, J.-P. (1980). Iterated Maps on the Interval as Dynamical Systems. Birkhauser.

[6] Oseledec, V. I. (1968). A multiplicative ergodic theorem: Lyapunov characteristic numbers for dynamical systems. Trans. Moscow Math. Soc., 19, 197–231.

[7] Benettin, G., Galgani, L., Giorgilli, A., & Strelcyn, J.-M. (1980). Lyapunov characteristic exponents for smooth dynamical systems. Meccanica, 15, 9–20.

[8] Kaneko, K. (1989). Spatiotemporal chaos in coupled map lattices. Physica D, 37(1–3), 60–82.

[9] Kaneko, K. (Ed.). (1993). Theory and Applications of Coupled Map Lattices. Wiley.

[10] Mandelbrot, B. B. (1980). Fractal aspects of the iteration of z -> z^2 + c. Annals of the New York Academy of Sciences, 357, 249–259.

[11] Mandelbrot, B. B. (1982). The Fractal Geometry of Nature. W. H. Freeman.

[12] Prigogine, I., & Nicolis, G. (1977). Self-Organization in Nonequilibrium Systems. Wiley.

[13] Prigogine, I., & Stengers, I. (1984). Order Out of Chaos. Bantam.

[14] Mac Lane, S., & Moerdijk, I. (1992). Sheaves in Geometry and Logic: A First Introduction to Topos Theory. Springer.

[15] The Univalent Foundations Program (2013). Homotopy Type Theory: Univalent Foundations of Mathematics. Institute for Advanced Study.

[16] Voevodsky, V. (2014). An experimental library of formalized mathematics based on the univalent foundations. Mathematical Structures in Computer Science, 25(5), 1278–1294.

[17] Lawvere, F. W. (2007). Axiomatic cohesion. Theory and Applications of Categories, 19(3), 41–49.

[18] Baez, J. C., & Fong, B. (2015). A compositional framework for passive linear networks. Theory and Applications of Categories, 33, 1158–1222.

[19] Fong, B. (2016). Decorated cospans. Theory and Applications of Categories, 30, 1096–1120.

[20] Baez, J. C., Courser, K., & Vaghese, C. (2020). Structured cospans. Theory and Applications of Categories, 35, 1771–1822.

[21] Baez, J. C., & Fong, B. (2015). A Noether theorem for Markov processes. Journal of Mathematical Physics, 56, 013301.

[22] Lasota, A., & Mackey, M. C. (1994). Chaos, Fractals, and Noise (2nd ed.). Springer.

[23] Pesin, Y. B. (1977). Characteristic Lyapunov exponents and smooth ergodic theory. Russian Mathematical Surveys, 32(4), 55–114.

[24] Curry, J. (2014). Sheaves, cosheaves and applications. arXiv:1303.3255.

[25] Robinson, A. (2014). Topological Signal Processing. Springer.

[26] Devaney, R. L. (1989). An Introduction to Chaotic Dynamical Systems (2nd ed.). Addison–Wesley.

[27] Beck, C., & Schlögl, F. (1993). Thermodynamics of Chaotic Systems. Cambridge University Press.

[28] Eckmann, J.-P., & Ruelle, D. (1985). Ergodic theory of chaos and strange attractors. Rev. Mod. Phys., 57(3), 617–656.

[29] Cvitanovic, P. (1991). Invariant measurement of strange sets in terms of cycles. Phys. Rev. Lett., 61, 2729–2732.

[30] Fatou, P. (1919–1920) and Julia, G. (1918–1919). Original papers on iteration of rational functions (foundational sources for Julia sets).

Source Code

```
#!/usr/bin/env python3
# Eclipse Topos — Figure Generator
# Requirements: Python 3.x, numpy, matplotlib
# Usage: python eclipse_topos_figures.py
# Notes:
#   - One chart per figure, matplotlib only, no explicit colors/styles.
#   - By default, the script both shows figures on screen and saves PNGs.
#   - All math/comments are ASCII-only.

import numpy as np
import matplotlib.pyplot as plt

# ---------------------------
# Global constants and helpers
# ---------------------------

def sech(z):
    return 1.0 / np.cosh(z)

LN2 = np.log(2.0)     # lambda_map(4) = ln(2)
LAMBDA0 = 0.10        # edge sharpness for complexity; tweak to make rim thinner/thicker
EPS_MAX = 0.36        # plotting radius
ETA = 1e-16           # tiny epsilon for logs
```

```
# High-res grids for heatmaps (Figs. 1, 4) and base fields
K = np.linspace(0.0, 2.0*np.pi, 1024)     # angle
E = np.linspace(0.0, EPS_MAX, 720)        # radius
KK, EE = np.meshgrid(K, E)

# Transverse exponent at r = 4:
# lambda_perp(k, eps) = ln(2) + ln(|1 - eps + eps*cos(k)|)
def lambda_perp(k, eps):
    return LN2 + np.log(np.abs(1.0 - eps + eps*np.cos(k)) + ETA)

# Complexity functional at r = 4 (1D worst-mode form used for polar field via k):
# C(r, eps) = h_top(r) * sech( (lambda_map(r) + ln(|1 - 2*eps|)) / lambda0 )
# Here h_top(4) = ln(2).
def complexity_field(k, eps):
    # We display C as a function of (k, eps) by using the same k on the coupling term.
    # This keeps the brightest band aligned with the zero set of lambda_perp.
    lam = lambda_perp(k, eps)                  # full k-resolved transverse exponent
    # Replace ln(|1 - 2*eps|) by lam - ln(2) to keep the rim alignment in (k, eps).
    # That is equivalent to the closed form in Sec. 3 when evaluated on the worst mode.
    return LN2 * sech(lam / LAMBDA0)

# Edge (rim) locus: solve lambda_perp(k, eps) = 0 on the branch with (1 - eps + eps*cos k) > 0
# This gives eps = 1 / (2*(1 - cos k)), where finite and within plotting radius.
def rim_curve(k_array):
    den = 1.0 - np.cos(k_array)
    with np.errstate(divide='ignore', invalid='ignore'):
```

```
    eps_star = 0.5 / den
  # Keep only finite values in [0, EPS_MAX]
  eps_star = np.where((den > 1e-12) & (eps_star >= 0.0) & (eps_star <= EPS_MAX), eps_star, np.nan)
  return eps_star


# ---------------------------
# Figure 1: Base complexity universe (polar heatmap of C with rim E)
# ---------------------------

def fig1_base_complexity(save=True):
  C = complexity_field(KK, EE)
  eps_star = rim_curve(K)


  plt.figure(figsize=(8, 8))
  ax = plt.subplot(111, projection='polar')
  ax.pcolormesh(K, E, C, shading='auto')
  ax.plot(K, eps_star, linewidth=2.0)
  ax.set_title("Fig. 1 — Base Complexity Universe (polar heatmap of C with rim E)", pad=12)
  ax.set_rlabel_position(135)
  ax.set_rticks([]); ax.set_thetagrids([]); ax.grid(False)
  if save:
    plt.tight_layout()
    plt.savefig("fig1_base_complexity.png", dpi=300, bbox_inches="tight")
  plt.show()
```

```
# ---------------------------
# Figure 2: Rim as manifold (marker size ~ |d lambda_perp / d eps| on E)
# ---------------------------

def fig2_rim_sharpness(save=True):
    k2 = np.linspace(1e-3, 2.0*np.pi - 1e-3, 800)
    eps_star = rim_curve(k2)
    valid = ~np.isnan(eps_star)
    k2 = k2[valid]; eps_star = eps_star[valid]

    # d/d(eps) lambda_perp = d/d(eps) ln(|1 - eps + eps*cos k|)
    # = (cos k - 1) / (1 - eps + eps*cos k), evaluated on the rim
    denom = 1.0 - eps_star + eps_star*np.cos(k2)
    sharp = np.abs((np.cos(k2) - 1.0) / (denom + ETA))
    # Normalize marker sizes for visibility
    s = 10.0 + 190.0 * (sharp - sharp.min()) / (sharp.max() - sharp.min() + 1e-12)

    plt.figure(figsize=(8, 8))
    ax = plt.subplot(111, projection='polar')
    ax.scatter(k2, eps_star, s=s)
    ax.set_title("Fig. 2 — Rim as Manifold (marker size ~ |d lambda_perp / d eps| on E)",
pad=12)
    ax.set_rlabel_position(135)
    ax.set_rticks([]); ax.set_thetagrids([]); ax.grid(False)
    if save:
        plt.tight_layout()
```

```
        plt.savefig("fig2_rim_sharpness.png", dpi=300, bbox_inches="tight")
    plt.show()

# ---------------------------
# Figure 3: Euler skeleton on the unit circle with arrows z -> z^2
# ---------------------------

def fig3_euler_skeleton(save=True):
    npts = 32
    theta = np.linspace(0.0, 2.0*np.pi, npts, endpoint=False)
    z = np.column_stack([np.cos(theta), np.sin(theta)])
    z2 = np.column_stack([np.cos(2.0*theta), np.sin(2.0*theta)])
    dx = z2[:, 0] - z[:, 0]
    dy = z2[:, 1] - z[:, 1]

    plt.figure(figsize=(7.5, 7.5))
    ax = plt.subplot(111)
    t = np.linspace(0.0, 2.0*np.pi, 1000)
    ax.plot(np.cos(t), np.sin(t))
    ax.quiver(z[:, 0], z[:, 1], dx, dy, angles="xy", scale_units="xy", scale=1.0, width=0.003)
    ax.set_aspect("equal", "box")
    ax.set_xlim(-1.15, 1.15); ax.set_ylim(-1.15, 1.15)
    ax.set_title("Fig. 3 — Euler Skeleton (unit circle; arrows for z -> z^2)")
    ax.axis("off")
    if save:
        plt.tight_layout()
```

```
        plt.savefig("fig3_euler_skeleton.png", dpi=300, bbox_inches="tight")
    plt.show()

# ---------------------------
# Figure 4: Ordered vs chaotic regions — sign of lambda_perp, rim as exact separator
# ---------------------------

def fig4_sign_partition(save=True):
    Lam = lambda_perp(KK, EE)
    eps_star = rim_curve(K)

    plt.figure(figsize=(8, 8))
    ax = plt.subplot(111, projection='polar')
    ax.pcolormesh(K, E, Lam, shading='auto')
    ax.plot(K, eps_star, linewidth=2.0)
    ax.set_title("Fig. 4 — Ordered (lambda_perp<0) vs Chaotic (lambda_perp>0); Rim
Separates", pad=12)
    ax.set_rlabel_position(135)
    ax.set_rticks([]); ax.set_thetagrids([]); ax.grid(False)
    if save:
        plt.tight_layout()
        plt.savefig("fig4_sign_partition.png", dpi=300, bbox_inches="tight")
    plt.show()

# ---------------------------
# Figure 5: Complexity flow field — quiver of grad(C) mapped onto the disk
```

```
# ---------------------------

def fig5_flow_field(save=True):
    # Coarse grid for arrows (keeps figure clean)
    ks = np.linspace(0.0, 2.0*np.pi, 60)
    es = np.linspace(0.02, 0.32, 28)  # avoid 0 to keep 1/eps well-behaved
    KS, ES = np.meshgrid(ks, es)

    # Complexity on coarse grid
    Lam_s = lambda_perp(KS, ES)
    C_s = LN2 * sech(Lam_s / LAMBDA0)

    # Finite difference gradients in polar coords
    dk = ks[1] - ks[0]
    de = es[1] - es[0]
    dCdk = np.gradient(C_s, axis=1) / dk
    dCde = np.gradient(C_s, axis=0) / de

    # Map to Cartesian for quiver
    X = ES * np.cos(KS)
    Y = ES * np.sin(KS)

    ES_safe = np.where(ES < 1e-3, 1e-3, ES)
    dCdx = dCde*np.cos(KS) - (dCdk/ES_safe)*np.sin(KS)
    dCdy = dCde*np.sin(KS) + (dCdk/ES_safe)*np.cos(KS)
```

```
    # Background: full-resolution complexity field mapped to square via imshow
    C_bg = complexity_field(KK, EE)
    # Convert polar (k, eps) to Cartesian extent; we will show it as a square image
    extent = (-EPS_MAX, EPS_MAX, -EPS_MAX, EPS_MAX)

    plt.figure(figsize=(8, 8))
    ax = plt.subplot(111)
    ax.imshow(C_bg, extent=extent, origin='lower', aspect='equal')
    ax.quiver(X, Y, dCdx, dCdy, angles='xy', scale_units='xy', scale=8.0, width=0.002)
    ax.set_xlim(-EPS_MAX, EPS_MAX); ax.set_ylim(-EPS_MAX, EPS_MAX)
    ax.set_title("Fig. 5 — Complexity Flow Field (quiver of grad(C) on disk)")
    ax.axis('off')
    if save:
        plt.tight_layout()
        plt.savefig("fig5_flow_field.png", dpi=300, bbox_inches="tight")
    plt.show()

# ---------------------------
# Main
# ---------------------------

if __name__ == "__main__":
    # Toggle saving to files here (True/False). Figures always display.
    SAVE_FILES = True

    fig1_base_complexity(save=SAVE_FILES)
```

fig2_rim_sharpness(save=SAVE_FILES)

fig3_euler_skeleton(save=SAVE_FILES)

fig4_sign_partition(save=SAVE_FILES)

fig5_flow_field(save=SAVE_FILES)